input from the state transition circuitry 102. In the event that no hit was found by the associate operation the new entry line is marked as valid and a further new line is invalidated ready for a next write operation.

In FIG. 3. both the content addressable memory cells 510. 610. 710 and 810 and the data RAMs 518. 618. 718. 818 store full addresses. In the case of the CAM cell. by "full address" is intended the full fetch address of each instruction word. For the data RAM. the "full address" means the precise target address. or the offset from the address of the branch instruction to be executed.

In one embodiment however both the CAM cells and the data RAM store less than full addresses. Specifically. before a branch instruction is entered into the branch target buffer the most significant bits of the fetch address of the branch instruction are compared with the most significant bits of the target address of that instruction. This is done by a comparator circuit comparing addresses on output lines 900 and 908 from the circuit 112. If the contents of a selected number of the most significant bits of the fetch address differ from the contents of the same number of selected bits of the target address. no entry is made into the branch target buffer. If however the contents of the selected number of most significant bits are the same in both the fetch address and the target address—i.e. the result of taking the branch instruction would be to branch to a new fetch address which is in the same or an adjacent program segment to the fetch address of the instruction itself. then the lesser significant bits of the target address of that instruction are stored in the branch target buffer. Similarly only the same number of lesser significant bits of the branch instruction address are written from line 900 into the buffer 400.

In this arrangement. during instruction fetching. the full fetch address is applied to the branch target buffer. although only the relevant number of lesser significant bits of the fetch address are input over the instruction word address bus 500. It will be apparent to one skilled in the art that in this embodiment instructions from other program segments. i.e. instructions having most significant bits which differ from the most significant bit of the branch instruction which is stored may have identical lesser significant bits to those of the instructions stored. The consequence of this is that instructions may be falsely identified as being branch instructions. with consequent processing delays.

We claim:

1. A method of loading entries into a cache memory comprising a content addressable memory (CAM) portion having a plurality of lines coupled to a respective line of a data memory portion. which method comprises the steps of effecting in parallel an associate operation and a write operation with the same address data. selectively controlling validity of one or more lines in the cache. invalidating one line of the cache. writing data including an address to said one line at the same time as an associate operation is carried out using the same address. and controlling re-validation of said one line in dependence on the result of the associate operation. thereby validating said line only if no cache hit is found and thereby preventing a single address having two entries in the same cache.

2. A method according to claim 1 in which in response to a cache hit in one CAM line. data stored in that line of the data memory portion is modified by said write operation.

3. A method according to claim 1 in which an associate input of said CAM is connected to selection circuitry arranged to receive both write data and associate data. said method including operating said selection circuitry when a write operation is effected to select said write data for the associate input.

4. A method of operating a branch target buffer. comprising a cache memory with a CAM portion having a plurality of lines coupled to a respective line of a data memory portion. to prevent a single branch instruction being stored as a multiple entry in said buffer. which method comprises selectively controlling validity of one or more lines in the cache. invalidating one line of the cache. writing data including an address to said one line at the same time as an associate operation is carried out using the same address. and controlling re-validation of said one line in dependence on the result of the associate operation. thereby validating said line only if no cache hit is found.

5. A method according to claim 4 in which in response to a cache hit in CAM line. data stored in that line of the data memory portion is modified by said write operation.

6. A method according to claim 5 in which said CAM is used to store address data relating to a plurality of branch instructions and said data memory portion is used to store for each branch instruction in the cache. a target address for the branch instruction together with a prediction of whether the branch will be taken when the instruction is executed by a processor. said write operation effecting a modification of at least one of said target address or prediction in the event of a cache hit during said write operation.

7. A method according to claim 6 in which said data memory portion also holds a prediction strength value in addition to said prediction. and said write operation is effective to modify said prediction strength value in the event of a cache hit during the write operation.

8. A cache memory comprising a content addressable memory (CAM) portion having a plurality of lines coupled to a respective line of a data memory portion. said CAM portion having an associate input for inputting data to effect an associate operation and a write input for inputting data to effect a write operation. said associate input being connected to input selection circuitry to select either write data or associate data for connection to said associate input whereby an associate operation may be effected in parallel with a write operation using common address data. said memory further comprising line validation circuitry operable in response to a result of an associate operation to prevent data related to said common address being held in more than one line of said cache.

9. A cache memory according to claim 8 further comprising. a branch target buffer connected to said cache. said CAM portion holding address data relating to a plurality of branch instructions and said data memory portion holding for each branch instruction in the cache. a target address for said branch instruction together with a prediction of whether the branch will be taken when the instruction is executed by a processor.

10. A branch target buffer according to claim 9 in which said cache includes data modification circuitry operable in response to a write operation to modify data stored for a branch instruction located by a cache hit during a write operation.

11. A branch target buffer according to claim 10 in which said data memory portion includes storage circuitry for storing a prediction strength value in addition to said prediction for any branch instruction held in said cache. and said storage circuitry is responsive to said data modification circuitry to modify said prediction strength value in response to a write operation relating to a branch instruction in said buffer.

12. The cache memory of claim 9 further comprising a memory for holding a plurality of instructions. wherein said processor executes instructions sequentially and said branch target buffer predicts the target address for the branch instruction fetched from said memory for execution by said processor.

13. A method of loading entries into a cache memory comprising a content addressable memory (CAM) portion having a plurality of lines coupled to a respective line of a data memory portion, which method comprises the steps of effecting in parallel an associate operation and a write operation with the same address data, selectively controlling validity of one or more lines in the cache, invalidating one line of the cache, writing data including an address to said one line at the same time as an associate operation is carried out using the same address, and controlling re-validation of said one line in dependence on the result of the associate operation, thereby validating said line only if no cache hit is found.